Belief Functions Applied to Object Recognition and Object Classification

A. P. Dempster
Department of Statistics
Harvard University
dempster@fas.harvard.edu

Calvin W. F. Chiu
Department of Statistics
Harvard University
wchiu@fas.harvard.edu

January 2, 2004

Submitted to *International Journal of Intelligent Systems*

Abstract

We consider situations where each individual member of a defined object set is characterized uniquely by a set of variables, and we propose models and associated methods that *recognize* or *classify* a newly observed individual. Inputs consist of uncertain observations on the new individual and on a memory bank of previously identified individuals. Outputs consist of uncertain inferences concerning degrees of agreement between the new object and previously identified objects or object classes, with inferences represented by Dempster-Shafer belief functions. We illustrate the approach using models constructed from independent *simple support* belief functions defined on binary variables. In the case of object recognition, our models lead to marginal belief functions concerning how well the new object matches objects in memory. In the classification model, we compute beliefs and plausibilities that the new object lies in defined subsets of an object set. When regarded as similarity measures, our belief and plausibility functions can be interpreted as candidate membership functions in the terminology of fuzzy logic.

1 Introduction

The term *object* refers to a specific individual in a defined object set C. A set of variables $V_1, V_2, \ldots, V_K$ is assumed to possess defined values for each individual in C, with the understanding that exact values of $V_1, V_2, \ldots, V_K$, if known, would characterize an individual uniquely. In practice, such values are rarely known exactly, but instead are known up to measures of uncertainty, here represented by belief functions. This uncertain knowledge is assumed to be stored in a data bank that we call *memory* for the members $I_1, I_2, \ldots, I_N$ of a subset C_{mem} of C.

- The problem of *object recognition* is to draw uncertain inferences about whether a new, uncertainly observed, individual I_0 belongs to selected subsets of C_{mem}, including singleton subsets and C_{mem} itself.

- The problem of *object classification* (aka *supervised machine learning*) extends the object recognition model in three ways: first by specifying a partition of C into mutually exclusive subclasses $C_1, C_2, \ldots, C_M$, second by assuming that the memory objects $I_1, I_2, \ldots, I_N$ are distributed in a known way among the subclasses C_m where $m \in \{1, 2, \ldots, M\}$, and third by assuming that the memory objects within each subclass are a random sample from the population of objects associated with the subclass. The goal of object classification is to draw inferences about the unknown subclass C_0 of the new individual I_0. Addressing this goal requires drawing statistical inferences from the uncertainly observed random subsets of the memory objects concerning population distributions of $V_1, V_2, \ldots, V_K$ in each C_m.

Our models are postulated in Section 3 wherein $V_1, V_2, \ldots, V_K$ are assumed to be binary, represented by indicators taking the values 0 and 1, and the actual subclasses of the memory objects are assumed to be known. If there were no uncertainty in either I_0 or C_{mem}, then the input to analysis would be a vector of K indicators for I_0, and corresponding vectors for the I_n in C_{mem}. Perfect matching could then be performed, meaning that the I_n could be ordered, at least partially, by counting the number of matches with I_0 on the K variables. A *match* would mean agreement on *all* K variables, and objects with a single mismatch or only a few mismatches might also be interesting as near relatives. If one of the memory objects were a match, then C_0 would assume the label of the subclass to which the matching memory object belonged; otherwise C_0 could be determined by methods such as nearest neighbors and neural networks.[1] Our goal is to study situations where perfect measurement, hence perfect matching, is not achievable, using belief functions to represent uncertainty about underlying matches and uncertainty about C_0.

The numerical algorithms illustrated in this report restrict the components of the models to simple support belief functions. A *simple support belief function* on a binary state space assigns probability p to one singleton subset of the state space and complementary probability $1 - p$ to the full state space. Jacob Bernoulli[2] called this a *pure probability argument*, in contrast to the more familiar *mixed probability argument* where probabilities p and $1 - p$ are assigned to the two singleton subsets. In the case of object recognition, our models lead to a marginal belief function concerning the match or nonmatch of I_0 with any I_n. In particular, the *belief* and *plausibility* of a match are measures of *similarity* between the observations on I_0 and the observations on I_n. In an analogous way, we are led to compute beliefs and plausibilities that I_0 belongs to each of the subclasses $C_1, C_2, \ldots, C_M$ in the classification model. When regarded as similarity measures, our belief and plausibility functions fill the same role as *membership functions* that quantify a degree of membership of an object in a class of objects, thus suggesting a relationship between belief functions and Zadeh's fuzzy logic.[3]

2 Elements of Belief Function Theory

In belief function theory, the set of possible states of the objective world is formally modeled as a *state space*

$$\mathcal{S} = \{a_1, a_2, \ldots, a_J\},$$

where restriction to finite $\mathcal{S}$ is unnecessary in general, but adequate for present purposes, and serves to avoid extraneous mathematical details. The a_j are called *atoms* of $\mathcal{S}$; only one atom is assumed to be the "actual state". We refer to the subsets $\mathcal{T}$ of $\mathcal{S}$ as *statements*; the space of all possible statements, *statement space*. For the finite $\mathcal{S}$ considered here, the statement space consists of $2^J - 1$ statements, including $\mathcal{S}$ itself, but excluding the empty subset $\emptyset$. The statement $\mathcal{T}$ is verbalized as the assertion

"The actual state of $\mathcal{S}$ lies in $\mathcal{T}$."

In this language, the role of belief functions is to quantify the uncertainty of an idealized observer (IO) regarding whether the statement $\mathcal{T}$ is true or false.

The essence of a belief function is a *basic probability assignment* over the statement space. In what follows, the subscript u will index the nonempty subsets $\mathcal{T}_u$ of $\mathcal{S}$. The basic probabilities associated with the nonempty subsets $\mathcal{T}_u$ of the state space $\mathcal{S}$ will be denoted $m_{\mathcal{S}}(\mathcal{T}_u)$, or simply $m(\mathcal{T}_u)$ when the associated state space is understood, where

$$\sum_u m(\mathcal{T}_u) = 1 \quad \text{and} \quad m(\mathcal{T}_u) \geq 0 \quad \text{for all } u\,.$$

From a mathematical prospective, a belief function is specified by a standard probability measure constructed over the sample space of statements. When the set of statements $\mathcal{T}_u$ with $m(\mathcal{T}_u) > 0$ is restricted to singleton subsets of $\mathcal{S}$, a belief function effectively becomes indistinguishable from a standard probability model.

No single value is available in belief function theory to identify as the probability of $\mathcal{T}_u$. Instead a pair of values, belief and plausibility, is needed. *Belief*, or $\mathrm{BEL}(\mathcal{T}_u)$, is a "lower probability" that sums $m(\mathcal{T}_v)$ over $\mathcal{T}_v$ contained in $\mathcal{T}_u$. *Plausibility*, or $\mathrm{PL}(\mathcal{T}_u)$, is an "upper probability" that sums

$m(\mathcal{T}_v)$ over $\mathcal{T}_v$ having a nonempty intersection with $\mathcal{T}_u$. BEL($\mathcal{T}_u$) and PL($\mathcal{T}_u$) are formal measures of the IO's uncertainty about the truth of the statement $\mathcal{T}_u$. BEL simply means formal subject probability (FSP) that *must* be assigned to the truth of $\mathcal{T}_u$, while PL means FSP that *may* be assigned to the truth of $\mathcal{T}_u$. In practice, for a safe bet on the statement $\mathcal{T}_u$, the correct choice of betting probability is BEL($\mathcal{T}_u$), where the IO is assumed to be betting on the truth of $\mathcal{T}_u$. Similarly, $1 - \mathrm{PL}(\mathcal{T}_u)$ is the complementary safe betting probability of the IO's opponent on the other side of the bet. Thus, although both bettors are assumed to adopt the same belief function, there in general remains a spread between the amounts that the theory advises for safe wagers.

The preceding paragraphs define the raw materials that the operations of belief function calculus manipulate. The two fundamental operations are *propagation* of a belief function from one state space to a related state space, and *fusion* of two or more belief functions on a common state space to yield a combined belief function. In practice, propagation and fusion are used in tandem. It often happens that independent belief functions are constructed on different margins of a larger state space, and it is necessary to propagate from each such margin to a common state space before fusion can operate to combine the information. Formal definitions are indicated briefly below; illustrations will appear in later sections.

Propagation links partitions of $\mathcal{S}$. A *partition* (also called a *margin*) of $\mathcal{S}$ is a collection of subsets of $\mathcal{S}$ that are mutually exclusive and include all the atoms of $\mathcal{S}$. Formally we write

$$\Pi = \{\pi_1, \pi_2, \ldots, \pi_M\} \,,$$

where each π_m is a subset of $\mathcal{S}$. Shafer[4] also called Π a *coarsening* of $\mathcal{S}$ because knowing that the actual state a_n in $\mathcal{S}$ is in π_m for some $m \in \{1, 2, \ldots, m\}$ in effect means that Π is interpretable as a state space referring to the same small world as $\mathcal{S}$, but incapable of storing as much information about the actual state as does $\mathcal{S}$ (except in the trivial case $M = N$).

Extension and marginalization are the two basic propagation operations. Extension assumes a belief function on a margin $\mathcal{X}$ and defines an *extended* belief function on $\mathcal{S}$ by applying the basic probabilities of the marginal belief function to the cylinder statements in $\mathcal{S}$ that project into the corresponding marginal statements in $\mathcal{X}$. Given a belief function on $\mathcal{S}$, a *marginal* belief function on a margin $\mathcal{P}$ is defined by *projecting* each subset of $\mathcal{S}$ into a corresponding subset of $\mathcal{P}$ and defining basic probabilities for subsets of $\mathcal{P}$ by summing the original basic probabilities whose statements project into a common statement in $\mathcal{P}$. Note that extension followed by marginalization is an identity operation, but many different belief functions on $\mathcal{S}$ yield the same marginal belief function on $\mathcal{P}$, so that marginalization followed by extension typically is not an identity operation.

The general form of propagation starts from a belief function on a first margin Π_1, extends to the full space $\mathcal{S}$, and then marginalizes to a second margin Π_2. A *propagated* belief function thus constructed describes uncertainty about statements concerning Π_2 implied by the original belief function on Π_1.

Fusion goes by several different names. Shafer[4] called it *Dempster's rule of combination* from its original appearance in reference 5, and also called it a *direct sum operator*. It should perhaps be called the *product-intersection rule*[6] because it generalizes the multiplication of Bayesian probabilities of independent events and incorporates, in purely logical models reduces to, the intersection rule of Boolean logic. Fusion assumes *independence* of the component belief functions, meaning no interactions are allowed among the sources of evidence behind the components. To combine two belief functions, one first intersects pairs of statements from the two input belief functions to create new basic statements, and then assigns these statements the product of the basic probabilities associated with the corresponding input statements. Note that in general only input statements carrying nonzero basic probabilities need be intersected. Another detail is that typically more than one pair of intersected statements results in the same new basic statement, in which case the rule requires that the resulting products must be summed to yield the basic probability for that new basic statement.

3 Models for Object Recognition and Object Classification

As previewed in Section 1, there are N memory objects and a new object, each characterized by a common set of K binary variables. These may be visualized as forming a matrix with $N+1$ rows and K columns as shown in (1), where the first row represents characteristics of the new object I_0, the remaining rows correspond to the memory objects I_n, and the X_{nk} denote the binary variables, with n indexing objects and k indexing variables. All models in this paper assume a state space defined as the product space of $(N+1)K$ binary variables. The (full) state space thus has $2^{(N+1)K}$ atoms, while the statement space has $(2^{2^{(N+1)K}}-1)$ statements.

$$\begin{bmatrix} X_{01} & X_{02} & \cdots & X_{0K} \\ X_{11} & X_{12} & \cdots & X_{1K} \\ \vdots & \vdots & \ddots & \vdots \\ X_{N1} & X_{N2} & \cdots & X_{NK} \end{bmatrix} \tag{1}$$

Uncertain knowledge of the actual values of the X_{nk} is described by belief functions whose basic probabilities are

$$m_{X_{nk}}(\{0\}), \quad m_{X_{nk}}(\{1\}), \quad \text{and} \quad m_{X_{nk}}(\{0,1\}).$$

We assume in this paper that each of these belief functions is of the simple support variety, so either $m_{X_{nk}}(\{0\})$ or $m_{X_{nk}}(\{1\})$ equals 0. With each unknown X_{nk}, we associate a known binary quantity Y_{nk}, taking a value 0 or 1, and an associated p_{nk} that represents the nonzero $m_{X_{nk}}(\{Y_{nk}\})$; because $m_{X_{nk}}(\{1-Y_{nk}\})=0$, we have $m_{X_{nk}}(\{0,1\})=1-p_{nk}$. The pair (Y_{nk},p_{nk}) is therefore a convenient representation of the assumed uncertainty about X_{nk}. We call Y_{nk} the *nominal value* of X_{nk}, and p_{nk} is the belief that $X_{nk}=Y_{nk}$. We emphasize that the representation (Y_{nk},p_{nk}) can only be used for simple support belief functions on binary margins.

We assume all component belief functions are independent. This means that, when the $(N+1)K$ belief functions are extended to the full state space, they may then be fused by the product-intersection rule to yield a combined belief function over the full state space, thus opening the path to computing beliefs and plausibilities about hypothesized matches between the new object and the memory objects. Independence here may seem to be a strong assumption. Our model is a natural starting point, however, because the role of the belief function assigned to each X_{nk} is like a specification of measurement error for that particular X_{nk}, and, in statistical modeling, measurement errors are often assumed to be independent.

3.1 The Case $N=K=1$

The simplest object recognition problem is to draw inferences about whether a characteristic of I_0 matches the corresponding characteristic of I_1, or equivalently whether the actual value of X_{01} equals the actual value of X_{11}. Omitting the second subscript, the state space concerning the pair (X_0,X_1) is

$$\{(0,0),(0,1),(1,0),(1,1)\}\,.$$

An independence model for this state space is constructed from two extensions followed by a fusion operation. To illustrate, suppose that uncertain knowledge about the actual values of X_0 and X_1 is represented by the independent simple support belief functions (Y_0,p_0) and (Y_1,p_1), so that the basic probabilities for the X_0 state space are

$$m_{X_0}(\{Y_0\})=p_0 \quad \text{and} \quad m_{X_0}(\{0,1\})=1-p_0\,,$$

and those for the X_1 state space are

$$m_{X_1}(\{Y_1\})=p_1 \quad \text{and} \quad m_{X_1}(\{0,1\})=1-p_1\,.$$

Since the statements $\{Y_0\}$ and $\{0,1\}$ concerning X_0 are equivalent to respectively the statements $\{(Y_0,0),(Y_0,1)\}$ and $\{(0,0),(0,1),(1,0),(1,1)\}$ concerning (X_0,X_1), applying the same basic probabilities p_0 and $1-p_0$ to these equivalent statements defines the extension from the X_0 margin to the (X_0, X_1) state space. The same principle applies to the X_1 margin, leading to a second extension that assigns basic probabilities p_1 and $1 - p_1$ to the statements $\{(0, Y_1), (1, Y_1)\}$ and $\{(0, 0), (0, 1), (1, 0), (1, 1)\}$ concerning (X_0, X_1). Fusing the two extended belief functions over statements of the (X_0, X_1) state space yields the combined belief function in Table 1.

Table 1: A combined belief function for the (X_0, X_1) state space

$\mathcal{T}_u$	$m_{(X_0,X_1)}(\,\mathcal{T}_u\,)$
$\{(Y_0, Y_1)\}$	$p_0 p_1$
$\{(Y_0, 0), (Y_0, 1)\}$	$p_0(1 - p_1)$
$\{(0, Y_1), (1, Y_1)\}$	$(1 - p_0)p_1$
$\{(0, 0), (0, 1), (1, 0), (1, 1)\}$	$(1 - p_0)(1 - p_1)$

Suppose that the variable ζ_1 stores the result of matching the actual values of X_0 and X_1, where

$$\zeta_1 = \begin{cases} 0 & \text{if} \quad X_1 \neq X_0\,, \\ 1 & \text{if} \quad X_1 = X_0\,. \end{cases}$$

Since the actual values of the X_n are unknown, the actual match result is unknown, but we can associate ζ_1 with a known binary quantity Z_1 where

$$Z_1 = \begin{cases} 0 & \text{if} \quad Y_1 \neq Y_0\,, \\ 1 & \text{if} \quad Y_1 = Y_0\,. \end{cases} \tag{2}$$

Like Y_0 and Y_1 being the nominal values of X_0 and X_1, Z_1 is the nominal value of ζ_1. Uncertain knowledge of the actual value of ζ_1 may be derived from uncertain knowledge of the actual values of X_0 and X_1 by marginalizing the combined belief function in Table 1 to the ζ_1 state space. The marginalization can be performed in two steps: first by projecting the four statements concerning (X_0, X_1) to the ζ_1 margin, so that the statement $\{(Y_0, Y_1)\}$ concerning (X_0, X_1) becomes the statement $\{Z_1\}$ concerning ζ_1, and the three statements $\{(Y_0, 0), (Y_0, 1)\}$, $\{(0, Y_1), (1, Y_1)\}$, and $\{(0, 0), (0, 1), (1, 0), (1, 1)\}$ concerning (X_0, X_1) all become the statement $\{0, 1\}$ concerning ζ_1, and second by assigning the basic probability $p_0 p_1$ to the statement $\{Z_1\}$ and the sum of the basic probabilities

$$p_0(1 - p_1), \quad (1 - p_0)p_1, \quad \text{and} \quad (1 - p_0)(1 - p_1)$$

to the statement $\{0, 1\}$. The resulting belief function for the ζ_1 margin is thus characterized by the basic probabilities

$$m_{\zeta_1}(\,\{Z_1\}\,) = p_0 p_1 \quad \text{and} \quad m_{\zeta_1}(\,\{0, 1\}\,) = 1 - p_0 p_1\,,$$

from which we computed

$$\mathrm{BEL}_{\zeta_1}(\{Z_1\}) = p_0 p_1 \quad \text{and} \quad \mathrm{PL}_{\zeta_1}(\{Z_1\}) = 1\,,$$

the belief and plausibility that the actual value of ζ_1 equals Z_1.

3.2 The Case of General N and $K = 1$

Generalizing the case of one memory object to the case of N memory objects, this section outlines the steps leading to inferences about whether the characteristic of I_0 matches the corresponding characteristics of the I_n in C_{mem}. The state space here consists of all 2^{N+1} possible realizations of the tuple $(X_0, X_1, \ldots, X_N)$. As in the previous section, uncertain knowledge about the actual values of the X_n is represented by the independent simple support belief functions (Y_n, p_n), which may be extended to the $(X_0, X_1, \ldots, X_N)$ state space and fused by the product-intersection rule to yield a combined belief function.

The results of matching the actual value of X_0 and the actual values $(X_1, \ldots, X_N)$ are stored in the vector $(\zeta_1, \ldots, \zeta_N)$ whose actual value is unknown because the actual values of the X_n are unknown. Each of the ζ_n is associated with a nominal value Z_n, similar to the Z_1 defined in (2). Uncertain knowledge of the actual value of $(\zeta_1, \ldots, \zeta_N)$ is described by the belief function marginalized from the combined belief function for the $(X_0, X_1, \ldots, X_N)$ state space. This marginal belief function may then be used to compute the beliefs and plausibilities for statements concerning $(\zeta_1, \ldots, \zeta_N)$ such as the singleton statement $\{(0, 1, 0, \ldots, 0)\}$ which corresponds to the assertion

"The actual value of X_0 matches the actual value of X_2 only,"

and the singleton statement $\{(0, 0, 1, \ldots, 1)\}$ which corresponds to the assertion

"The actual value of X_0 matches the actual values of all the X_n, except X_1 and X_2."

3.3 Object Recognition with General N and K

As mentioned at the beginning of Section 3, the state space for the case of general N and K has $2^{(N+1)K}$ atoms, each corresponding to a realization of the $(N + 1) \times K$ matrix in (1). Suppose that the results of matching a characteristic of I_0 and the corresponding characteristics of the I_n in C_{mem}

are stored in a column of the $N \times K$ matrix

$$\begin{bmatrix} \zeta_{11} & \zeta_{12} & \cdots & \zeta_{1K} \\ \zeta_{21} & \zeta_{22} & \cdots & \zeta_{2K} \\ \vdots & \vdots & \ddots & \vdots \\ \zeta_{N1} & \zeta_{N2} & \cdots & \zeta_{NK} \end{bmatrix} . \tag{3}$$

A row of ones indicates agreement on all K characteristics, meaning that the corresponding memory object is a match. Formally we let

$$\alpha_n = \begin{cases} 0 & \text{if} \quad \zeta_{nk} = 0 \text{ for some } k\,, \\ 1 & \text{if} \quad \zeta_{nk} = 1 \text{ for all } k\,, \end{cases}$$

where $k \in \{1, 2, \ldots, K\}$. The mathematical assertion $\alpha_n = 1$ thus represents the scientific statement

"The memory object I_n is a match,"

while $\alpha_n = 0$ represents "the memory object I_n is not a match."

Similar to Sections 3.1 and 3.2, uncertain knowledge about the actual values of the X_{nk} is described by the simple support belief functions (Y_{nk}, p_{nk}), assumed to be independent within and across individuals. Following the steps outlined in Section 3.2, we first obtain a combined belief function for the state space concerning a column of the characteristics $(X_{0k}, X_{1k}, \ldots, X_{Nk})$ in (1) and then marginalize it to yield a belief function for the state space concerning the corresponding column of the match results $(\zeta_{1k}, \ldots, \zeta_{Nk})$ in (3). Repeating these steps for the K variables generates K belief functions, one for each column in (3). These K belief functions are independent and therefore may be extended to the state space concerning (3) where the extended belief functions are fused to produce a combined belief function, which is then marginalized to the state space concerning $(\alpha_1, \ldots, \alpha_N)$, leading to a belief function representing uncertain knowledge of the actual value of $(\alpha_1, \ldots, \alpha_N)$, from which the beliefs and plausibilities for statements concerning $(\zeta_1, \ldots, \zeta_N)$ can be computed. Interesting statements include, for example, the singleton statement $\{(0, \ldots, 0)\}$ which corresponds to the assertion

"None of the memory objects is a match,"

or sometimes simply

"None of the above",

and the singleton statement $\{(1, 0, \ldots, 0)\}$ which corresponds to the assertion

"Only the memory object I_1 is a match."

3.4 Extension to Object Classification

The memory objects within each subclass constitute a *training set* in machine learning terminology. Each training set is regarded as a representative sample from the population of objects associated with its subclass. The goal of object classification is to draw inferences about C_0. While achieving this goal in principle includes drawing statistical inferences about the underlying class populations based on the uncertain training data, this difficult task may be finessed when the training sets are all sufficiently large by simply approximating the population distributions in nonparametric fashion by their corresponding training set distributions. For illustrative purposes only, we assume in Section 4.4 that our small samples adequately represent corresponding populations. With this simplification, the class recognition model becomes part of the object recognition model.

4 Numerical Examples

In this section, we assume that

- there exits a object set C that can be partitioned into two mutually exclusive subclasses C_1 and C_2;
- there is a memory C_{mem} that has $N = 7$ members $I_1, I_2, \ldots, I_7$;
- the memory objects I_1, I_2, I_3, and I_4 belong to C_1, while I_5, I_6, and I_7 belong to C_2;
- there is a new object I_0 whose subclass C_0 is unknown;
- all 8 objects are characterized by a set of $K = 4$ binary variables whose actual values are unknown;
- uncertain knowledge of the actual values is represented by the $(N+1)K = 32$ simple support belief functions (Y_{nk}, p_{nk}), where the nominal values Y_{nk} are displayed in Table 2, and the beliefs p_{nk} will be specified later in the examples;
- all 32 belief functions are independent and thus may be extended from the binary margins to the full state space where the extended belief functions are fused by the product-intersection rule to yield a combined belief function.

The goal is to draw uncertain inferences about whether any of the I_n is a match for I_0 and uncertain inferences about the actual value of C_0.

TABLE 2 HERE

4.1 Two Special Cases

This example illustrates two special cases: *case 1* where all the p_{nk} equal 0.999, and *case 2* where all the p_{nk} equal 0.001. In each of these cases, we followed the steps outlined in Section 3.3 and obtained a combined belief function for the $(\alpha_1, \alpha_2, \ldots, \alpha_7)$ state space, from which we computed various beliefs and plausibilities about matches between I_0 and the I_n. The results are summarized in Table 3.

TABLE 3 HERE

In case 1, the belief and plausibility that "I_1 is a match" are 0.9920 and 1, strongly identifying the new object as I_1, which would have been the conclusion if all the nominal values in Table 2 were in fact the actual values. The tiny plausibilities for the second to the last statements effectively rule out $I_2, I_3, \ldots, I_7$ as potential matches and the possibility of finding no match in the memory. Note that the difference between the belief and the plausibility for each statement is close to 0, a property associated with inferences based on observations with little uncertainty.

In case 2, the difference between the belief and the plausibility for each of the first seven statements is almost 1, and for the last statement is 1, reflecting almost *total ignorance* about the listed statements, an effect of attempting to draw inferences from very uncertain observations.

4.2 When I_0 is "Blurred"

This example studies the cases where some characteristics of I_0 are quite uncertainly observed. Table 4 displays the results for two cases: *case 3* where $p_{04} = 0.5$ and all other p_{nk} equal 0.99, and *case 4* where $p_{03} = p_{04} = 0.5$ and all other p_{nk} equal 0.99.

TABLE 4 HERE

In case 3, the small plausibilities rule out $I_2, I_3, \ldots, I_7$ as potential matches. Similarly, I_2, I_3, I_5, I_6, and I_7 are ruled out as potential matches in case 4. Notice that, in both cases, we have exactly

$$\mathrm{BEL}(\text{ "}I_1\text{ is a match" }) = 1 - \mathrm{PL}(\text{ "None of the above" })\,.$$

This is just a coincidence because it happens in this example that

$$\mathrm{BEL}(\text{ "}I_1\text{ is a match" }) = \mathrm{BEL}(\text{ not "None of the above" })\,.$$

4.3 When No Memory Object is a Match

Suppose that the nominal values for I_0 are (0 0 0 0) instead of (0 0 0 1), so that no memory object would match the new object if all the nominal values were in fact the actual values. We assume here that $p_{nk} = 0.99$ for all n and k. The second and the third columns of Table 5 list the beliefs and plausibilities for some statements concerning $(\alpha_1, \alpha_2, \ldots, \alpha_7)$. The small plausibilities for the first seven statements rule out all memory objects as potential matches. The belief and plausibility for "none of the above" are 0.9411 and 1, strongly supporting the conclusion that none of the memory objects is a match for I_0.

TABLE 5 HERE

Excluding the "none of the above" option, we re-computed the beliefs and plausibilities. The forth and fifth columns of Table 5 display the results, which indicate that I_1, I_2, and I_3 are the best potential matches among the memory objects, with I_3 being the first choice, since the associated belief is slightly larger. At first glance, one would expect I_1, I_2, and I_3 to be equally likely potential matches, because each has three nominal values matching those of I_0. Nonetheless, the model chooses I_3 as the best potential match. In fact, the exact order from the best to the least potential matches is I_3, I_1, I_2, I_4, I_5, I_6, and I_7. Why is I_3 a winner? The answer is that I_3 looks "most unlike" the nonmatching memory objects—for example, when the third variable in Table 2 is excluded, I_1 and I_4 have exactly the same nominal values, so do I_2 and I_5. Note that the results in Table 5 suggest that both I_4 and I_5 are nonmatches. The model accounts for the fact that parts of I_1 and I_2 look very much like I_4 and I_5 who (as a "whole") look quite unlike I_0, and hence favors I_3 over I_1 and I_2.

Now we modify the nominal values for I_4 and I_5, so that they will no longer look like I_1 and I_2. Suppose that the nominal values for I_4 are now (1 0 1 1) instead of (0 0 1 1), and that the nominal values for I_5 are now (0 1 1 1) instead of (0 1 1 0). Table 6 displays the modified nominal values considered here, including the change from (0 0 0 1) to (0 0 0 0) for I_0. Keeping $p_{nk} = 0.99$ for all n and k, we calculated the beliefs and plausibilities from the model constructed from the simple support belief functions characterized by the modified nominal values and the p_{nk}. Table 7 shows the results, which recommend equally I_1, I_2, and I_3 to be the best potential matches for I_0.

TABLE 6 HERE

TABLE 7 HERE

Table 8 summarizes the results for the case where $p_{04} = 0.5$ while all other p_{nk} equal 0.99. If "none of the above" is an option, the model will say so, because, among the listed statements, "none of the above" is the only statement that has nonzero belief (0.4755) and plausibility 1. If the model must select a candidate from the memory, it will recommend I_1, because the associated belief and plausibility become 0.9245 and 0.9628, when "none of the above" is excluded as a possible answer. In both cases, the model strongly suggests that $I_2, I_3, \ldots, I_7$ are nonmatches.

TABLE 8 HERE

4.4 From Recognition to Classification

For the moment, we treat the nominal values in Table 2 as the actual values and examine the type of information that each of the four variables has regarding the actual value of C_0. Since the first variable of I_0 has value that matches the values of the corresponding variable of both the C_1-subclass objects $\{I_1, I_2, I_3\}$ and the C_2-subclass object I_5, the first variable carries no specific information as to whether $C_0 = C_1$ or $C_0 = C_2$. Similarly, the fourth variable says nothing specific about the actual value of C_0. The values of the second and the third variables of I_0 match those of the corresponding variables of some C_1-subclass objects but not any of the C_2-subclass objects. Thus, each of the second and the third variables carries some information supporting $C_0 = C_1$ but no information supporting $C_0 = C_2$.

Formally the state space concerning C_0 is the two-element set $\{C_1, C_2\}$. Assuming that the p_{nk} are all 0.99, we can then obtain four independent belief functions for the C_0 state space, each marginalized from the combined model for the state space concerning $(\zeta_{1k}, \ldots, \zeta_{7k})$, the match results for the kth variable, where $k \in \{1, 2, 3, 4\}$ indexes the four variables in Table 2. Associated with the first and the fourth variables, the marginal belief functions for the C_0 state space are both of the *vacuous* variety, each taking the form

$$m_{C_0}(\,\{C_1, C_2\}\,) = 1\,,$$

while the marginal belief functions associated with the second and the third variables are both of the simple support variety, each taking the form

$$m_{C_0}(\,\{C_1\}\,) = 0.9606 \quad \text{and} \quad m_{C_0}(\,\{C_1, C_2\}\,) = 0.0394\,.$$

Fusing these four marginal belief functions yields a combined belief function for the C_0 state space whose basic probabilities are

$$m_{C_0}(\{C_1\}) = 0.9984 \quad \text{and} \quad m_{C_0}(\{C_1, C_2\}) = 0.0016\,. \tag{4}$$

Thus, the belief that $C_0 = C_1$ is 0.9984. This may be interpreted as a membership value in the terminology of fuzzy logic that quantifies the degree to which I_0 belongs to the subclass C_1.

We emphasize that if the nominal values for I_0 were (0 0 0 0) instead of (0 0 0 1), in which case the recognition model would fail to find a match from among the memory objects (as illustrated in Section 4.3), but the combined belief function for the C_0 state space would be exactly the same as that in (4), leading to a conclusion with very high belief that I_0 belongs to the subclass C_1.

5 Summary

Using independent simple support belief functions as building blocks, we constructed new models for object recognition and object classification. We illustrated through numerical examples the performance of our models in various situations. When there is little uncertainty in both the observations on the new object and the memory objects, the recognition model will find a match, if there is one, in the memory with high belief. For the cases in which no memory object actually matches the new object, the recognition model will suggest that all memory objects are nonmatches and return the verdict "none of the above", while the classification model will often be able to assign the new object to a subclass. If "none of the above" were not an option, the recognition model would identify the memory objects that were "most similar" to the new object. In situations where some observations of the new object are quite uncertain, the models appear to perform well in ruling out many nonmatches, leaving a manageable number of potential matches for further analyses. Our classification model ouputs the beliefs that the new object belongs to each of the possible subclasses; these beliefs can be interpreted as membership values that quantify the degrees to which the new object belongs to each of the subclasses, thus suggesting a relationship between belief functions and fuzzy logic.

Acknowledgement

We are very grateful for the financial support provided by the Office of Naval Research Contract N00014-02-1-0412.

References

1. Bishop, C. M. *Neural Networks for Pattern Recognition;* Oxford University Press: New York, 1995.

2. Bernoulli, J. *Ars Conjectandi;* Basel, 1713.

3. Dempster, A. P. "Commentary on 'Membership Functions and Probability Measures of Fuzzy Sets' by Nozer D. Singpurwalla and Jane M. Booker", To appear in *Journal of the American Statistical Association*, 2004.

4. Shafer, G. *A Mathematical Theory of Evidence;* Princeton: New Jersey, 1976.

5. Dempster, A. P. *Annals of Mathematical Statistics* **1967,** *38*, 325–339.

6. Dempster, A. P. Construction and Local Computation Aspects of Network Belief Functions. In *Influence Diagrams, Belief Nets and Decision Analysis*; Oliver, R. M.; Smith, J. Q., Eds.; John Wiley: Chichester, 1990.

Table 2: The subclasses and the nominal values for $I_0, I_1, \ldots, I_7$.

Object	**Subclass**	**Variable**			
I_0	C_0	0	0	0	1
I_1	C_1	0	0	0	1
I_2		0	1	0	0
I_3		1	0	0	0
I_4		0	0	1	1
I_5	C_2	0	1	1	0
I_6		1	1	1	0
I_7		1	1	1	1

Table 3: Beliefs and plausibilities for the two special cases

Statement	**Case 1** $p_{nk} = 0.999 \quad \forall\, n, k$		**Case 2** $p_{nk} = 0.001 \quad \forall\, n, k$	
	BEL	PL	BEL	PL
I_1 is a match	0.9920	1	1×10^{-24}	1
I_2 is a match	0	3.0×10^{-6}	0	0.999998
I_3 is a match	0	3.0×10^{-6}	0	0.999998
I_4 is a match	0	1.0×10^{-3}	0	0.999999
I_5 is a match	0	7.0×10^{-9}	0	0.999997
I_6 is a match	0	1.5×10^{-11}	0	0.999996
I_7 is a match	0	7.0×10^{-9}	0	0.999997
None of the above	0	8.0×10^{-3}	0	1

Table 4: Beliefs and plausibilities for cases where I_0 is "blurred"

Statement	**Case 3**: $p_{04} = 0.5$ and all other p_{nk} equal 0.99		**Case 4**: $p_{03} = p_{04} = 0.5$ all other p_{nk} equal 0.99	
	BEL	PL	BEL	PL
I_1 is a match	0.4660	1	0.2354	1
I_2 is a match	0	0.0100	0	0.0100
I_3 is a match	0	0.0100	0	0.0100
I_4 is a match	0	0.0152	0	0.5026
I_5 is a match	0	0.0002	0	0.0051
I_6 is a match	0	3.98×10^{-6}	0	0.0001
I_7 is a match	0	7.41×10^{-6}	0	0.0002
None of the above	0	0.5340	0	0.7646

Table 5: When the nominal values for I_0 are modified from (0 0 0 1) to (0 0 0 0)

Statement	$p_{nk} = 0.99 \quad \forall\, n, k$		Excluding "none of the above"	
	BEL	PL	BEL	PL
I_1 is a match	0	0.0199	0.3212	0.3379
I_2 is a match	0	0.0199	0.3212	0.3379
I_3 is a match	0	0.0199	0.3244	0.3379
I_4 is a match	0	0.0004	0.0032	0.0067
I_5 is a match	0	0.0004	0.0032	0.0067
I_6 is a match	0	7.88×10^{-6}	3.1×10^{-5}	0.0001
I_7 is a match	0	1.57×10^{-7}	3.1×10^{-7}	2.7×10^{-6}
None of the above	0.9411	1		

Table 6: The modified nominal values for I_0, I_4, and I_5

Object	**Subclass**	**Variable**			
I_0	C_0	0	0	0	0
I_1	C_1	0	0	0	1
I_2		0	1	0	0
I_3		1	0	0	0
I_4		1	0	1	1
I_5	C_2	0	1	1	1
I_6		1	1	1	0
I_7		1	1	1	1

Table 7: Results for the analyses with the modified nominal values as input

Statement	$p_{nk} = 0.99 \quad \forall\, n, k$		Excluding "none of the above"	
	BEL	PL	BEL	PL
I_1 is a match	0	0.0199	0.3266	0.3400
I_2 is a match	0	0.0199	0.3266	0.3400
I_3 is a match	0	0.0199	0.3266	0.3400
I_4 is a match	0	7.88×10^{-6}	3.20×10^{-5}	0.0001
I_5 is a match	0	7.88×10^{-6}	3.20×10^{-5}	0.0001
I_6 is a match	0	7.88×10^{-6}	3.20×10^{-5}	0.0001
I_7 is a match	0	1.57×10^{-7}	3.20×10^{-7}	2.7×10^{-6}
None of the above	0.9415	1		

Table 8: Results for $p_{04} = 0.5$ and all other $p_{nk} = 0.99$

Statement	$p_{04} = 0.5$ and all other $p_{nk} = 0.99$		Excluding "none of the above"	
	BEL	PL	BEL	PL
I_1 is a match	0	0.5050	0.9245	0.9628
I_2 is a match	0	0.0199	0.0184	0.0379
I_3 is a match	0	0.0199	0.0184	0.0379
I_4 is a match	0	0.0002	1.80×10^{-6}	0.0004
I_5 is a match	0	0.0002	1.80×10^{-6}	0.0004
I_6 is a match	0	7.88×10^{-6}	1.80×10^{-6}	1.50×10^{-5}
I_7 is a match	0	3.98×10^{-6}	1.80×10^{-8}	7.56×10^{-6}
None of the above	0.4755	1		